```
<!ATTLIST ex.scsi.connector
   %common.attrib;
>

<!ELEMENT serial.ports (#PCDATA)>
<!ATTLIST serial.ports
   %common.attrib;
>

<!ELEMENT inf.serial.ports (#PCDATA)>
<!ATTLIST inf.serial.ports
   %common.attrib;
>

<!ELEMENT parallel.port (A | B | C )>
<!ATTLIST parallel.port
   %common.attrib;
>

<!ELEMENT extended.parallel.protocols (other?, A?, B?, C?)>
<!ATTLIST extended.parallel.protocols
   %common.attrib;
>

<!ELEMENT keyboard.mouse.port.type (other | A | B | C )>
<!ATTLIST keyboard.mouse.port.type
   %common.attrib;
>

<!ELEMENT mouse.port.type (other | A | B | C )>
<!ATTLIST mouse.port.type
   %common.attrib;
>

<!ELEMENT other.port.set (other.port+)>
<!ATTLIST other.port.set
   %common.attrib;
>

<!ELEMENT other.port (#PCDATA)>
<!ATTLIST other.port
   %common.attrib;
>

<!ELEMENT accessory (accessory.purpose,
   accessory.replaces, accessory.works.with)>
<!ATTLIST accessory
   %common.attrib;
>

<!ELEMENT accessory.purpose (#PCDATA)>
<!ATTLIST accessory.purpose
   %common.attrib;
>

<!ELEMENT accessory.replaces (#PCDATA)>
<!ATTLIST accessory.replaces
```

```
   %common.attrib;
>

<!ELEMENT accessory.works.with (#PCDATA)>
<!ATTLIST accessory.works.with
   %common.attrib;
>
```

```
<!-- isrvprim.mod   Version:  0.1 -->
<!-- Purpose:  provide primitives for service descriptions -->
<!-- Terry Allen  2 Jan 1998 -->
<!-- Copyright 1998 CNgroup, Inc. -->

<!ELEMENT service (service.name,
   service.function.sequence+, service.location.pointer*,
   contact*)>
<!ATTLIST service
   %common.attrib;
>

<!ELEMENT service.name (#PCDATA)>
<!ATTLIST service.name
   %common.attrib;
>

<!ELEMENT service.function.sequence (service.function+)>
<!ATTLIST service.function.sequence
   %common.attrib;
>

<!ELEMENT service.function (doctype+, service.location*)>
<!ATTLIST service.function
   %common.attrib;
>

<!ELEMENT doctype (#PCDATA)>
<!ATTLIST doctype
   %common.attrib;
   %party.attrib;
   %from.party.attrib;
   %to.party.attrib;
>

<!ELEMENT action EMPTY>
<!ATTLIST action
   verb (register | retrieve | query | unregister | notarize
      | act.upon ) #REQUIRED
>

<!ELEMENT service.location.pointer (%xll.or.urn;)>
<!ATTLIST service.location.pointer
   %common.attrib;
   cblpointer  CDATA    #FIXED      "outside"
>


<!ELEMENT contact (contact.function*, personal.name*,
   language.understood*,
   occupation.title?, occupation.code?, address.set+)>
<!ATTLIST contact
   %common.attrib;
>

<!ELEMENT contact.function (#PCDATA)>
<!ATTLIST contact.function
```

```
   %common.attrib;
>

<!ELEMENT language.understood EMPTY>
<!ATTLIST language.understood
   %lang.attrib.required;
>

<!ELEMENT service.set (service+)>
<!ATTLIST service.set
>
```

```
<!-- itaxo.dtd   Version:  0.1 -->
<!-- Purpose:  define taxonomy structure for Ingram Micro demo -->
<!-- Terry Allen  5 Jan 1998 -->
<!-- Copyright 1998 CNgroup, Inc. -->

<!ENTITY % common SYSTEM "icommatt.mod">
%common;

<!ENTITY % pointers SYSTEM "ipointer.mod">
%pointers;

<!ELEMENT taxon (taxon.name, taxon.id,
      taxon.info?, taxon.parent.pointer*,
      (taxon.child.pointer | taxon)*)>
<!ATTLIST taxon
   %common.attrib;
>

<!ELEMENT taxon.name (#PCDATA)>
<!ATTLIST taxon.name
   %common.attrib;
>

<!ELEMENT taxon.id (#PCDATA)>
<!ATTLIST taxon.id
   %common.attrib;
>

<!ELEMENT taxon.info (#PCDATA)>
<!ATTLIST taxon.info
   %common.attrib;
>
```

```
<?xml version="1.0"?>
<!-- itaxo.xml Version:  0.2 -->
<!-- Purpose:  part of Ingram Micro taxonomy -->
<!-- Terry Allen  5 Jan 1998 -->
<!-- Copyright 1998 CNgroup, Inc. -->

<!DOCTYPE taxon SYSTEM "itaxo.dtd">
<taxon>
<taxon.name>Ingram Micro Taxonomy of Computer Goods
</taxon.name>
<taxon.id>ingram:root
</taxon.id>
<taxon.info>Everything Ingram Micro sells
</taxon.info>
   <taxon>
<taxon.name>Computer Systems
</taxon.name>
<taxon.id>ingram:00
</taxon.id>
<taxon.info>Computers and some components
</taxon.info>
      <taxon>
<taxon.name>Desktop Computers
</taxon.name>
<taxon.id>ingram:00.01
</taxon.id>
<taxon.info>Stub for first subdivision
</taxon.info>
</taxon>
      <taxon>
<taxon.name>Tower Computers
</taxon.name>
<taxon.id>ingram:00.03
</taxon.id>
<taxon.info>Stub for second subdivision, all other subdivisions
   omitted save for no. 11
</taxon.info>
</taxon>
      <taxon>
<taxon.name>Portable Computer, Memory &amp;amp; Accessories
</taxon.name>
<taxon.id>ingram:00.11
</taxon.id>
<taxon.info>Laptops, etc., not further categorized in this
   taxonomy
</taxon.info>
</taxon>
</taxon>
</taxon>
```

```
<!-- ithink.xml Version:  0.1 -->
<!-- Purpose:  inventory request for Ingram Micro demo -->
<!-- Terry Allen  5 Jan 1998 -->
<!-- Copyright 1998 CNgroup, Inc. -->

<?xml version="1.0"?>
<!DOCTYPE request.for.info SYSTEM "iireq.dtd">

<request.for.info>
<info.description.set>
<info.description>
<xml.descriptor>
<doctype><dtd systemid="iinv.dtd"/></doctype>
<xml.descriptor.details>
<xll.xptr.frag>ROOT()(1,line.description.one)STRING(ThinkPad 770)
</xll.xptr.frag>
<xll.xptr.frag>ROOT()(1,pentium-mmx)
</xll.xptr.frag>
</xml.descriptor.details>
</xml.descriptor>
</info.description>
</info.description.set>
</request.for.info>
```

```
<?xml version="1.0"?>
<!-- think.xml Version:  0.1 -->
<!-- Purpose:  IBM product description for Ingram Micro demo -->
<!-- Terry Allen  3 Jan 1998 -->
<!-- Copyright 1998 CNgroup, Inc. -->

<!DOCTYPE product.description SYSTEM "iprod.dtd">

<product.description>
 <meta>
  <urn>urn:x-cbl:ISBN%200-944940:test:companies:ibm:000001
  </urn>
 </meta>
 <general.product.info>
  <product.identity>

   <product.id assigned.by="manufacturer">TBS
   </product.id>
   <product.id assigned.by="ingram">
   </product.id>
   <ingram.taxonomy.category>ingram:00.11
   </ingram.taxonomy.category>

  </product.identity>

  <bundling.and.shipping>

   <open.branch.warehouse.number>
   </open.branch.warehouse.number>
   <case.pack.quantity>
   </case.pack.quantity>
   <pallet.quantity>
   </pallet.quantity>
   <weight.per.unit>
   </weight.per.unit>
   <unit.of.weight>
   </unit.of.weight>
   <shippable.carton.flag>
   </shippable.carton.flag>
   <product.sn.on.box yesorno="yes"/>
   <barcoded.sn.on.box yesorno="yes"/>
   <package.length>
   </package.length>
   <package.width>
   </package.width>
   <package.height>
   </package.height>
   <unit.of.size>
   </unit.of.size>
   <upc>
   </upc>
   <container.upc>
   </container.upc>
   <bill.of.materials>
   </bill.of.materials>
   <media.codes>
   </media.codes>
```

```
    <cpu.codes>
    </cpu.codes>
    <line.description.one>ThinkPad 770
    </line.description.one>
    <line.description.two>
    </line.description.two>
    <sales.description>
    </sales.description>
    <selling.bullet.set>
     <selling.bullet>
     </selling.bullet>
     <selling.bullet>
     </selling.bullet>
    </selling.bullet.set>
    <requirements.set>
    <requirement>
    </requirement>
    <requirement>
    </requirement>
    </requirements.set>
   </bundling.and.shipping>

  </general.product.info>

  <supplemental.product.info>
   <options.accessories>
   </options.accessories>
   <addon.consumables>
   </addon.consumables>
  </supplemental.product.info>

  <product.dimensions.and.warranty>

   <physical.dimensions>
    <product.length>
    </product.length>
    <product.width>
    </product.width>
    <product.height>
    </product.height>
    <product.weight>
    </product.weight>
   </physical.dimensions>

   <warranty.info>
    <standard.warranty>
    </standard.warranty>
    <onsite.support yesorno="yes"/>
    <onsite.yr2 yesorno="yes"/>
    <onsite.yr3 yesorno="yes"/>
    <next.day.support yesorno="yes"/>
    <remarks>
    </remarks>
   </warranty.info>

  </product.dimensions.and.warranty>
```

```
<product.specifications>

 <physical.specs>

  <system.type><A/>
  </system.type>
  <battery.type><B/>
  </battery.type>
  <est.batt.life>
  </est.batt.life>
  <batt.recharge.time>
  </batt.recharge.time>
  <number.batteries.included>
  </number.batteries.included>
  <max.no.batts.installable>
  </max.no.batts.installable>
  <power.management>
  </power.management>
  <energy.star yesorno="yes"/>
  <voltage.supported><C/>
  </voltage.supported>
  <ac.adapter.type><A/>
  </ac.adapter.type>
  <ac.adapter.capacity>
  </ac.adapter.capacity>
  <security.feature.set><A/>
  </security.feature.set>
  <pointing.device><A/>
  </pointing.device>
  <number.buttons.pd>
  </number.buttons.pd>
  <pd.location><A/>
  </pd.location>
  <docking.station>
  </docking.station>
  <os.set>
  <os>nt
  </os>
  <os>win95
  </os>
  <os>win95
  </os>
  </os.set>
  <system.license><A/>
  </system.license>
  <bundled.software>
   <software.product>
   </software.product>
  </bundled.software>
  <net.interface><B/>
  </net.interface>
  <net.management.included>
  </net.management.included>

 </physical.specs>

 <cpu>
```

```
<processor>
 <pentium-mmx/>
</processor>
<p.speed.set>
<p.speed>200
</p.speed>
<p.speed>233
</p.speed>
</p.speed.set>
<p.upgrade.method>
</p.upgrade.method>
<ram.installed>
</ram.installed>
<ram.max>
</ram.max>
<ram.type><other></other>
</ram.type>
<l1.cache>
</l1.cache>
<l2.cache>
</l2.cache>
<l2.type><B/>
</l2.type>
<bios.manufacturer><A/>
</bios.manufacturer>
<flash.upgrade.bios yesorno="yes"/>
<hard.disk.capacity.set>
<hard.disk.capacity>3.2
</hard.disk.capacity>
<hard.disk.capacity>4
</hard.disk.capacity>
<hard.disk.capacity>5.1
</hard.disk.capacity>
</hard.disk.capacity.set>
<h.d.interface><B/>
</h.d.interface>
<h.d.user.removable yesorno="no"/>
<removable.drive.set><A/>
</removable.drive.set>
<small.floppy.drive.type><A/>
</small.floppy.drive.type>
<cdrom.speed><C/>
</cdrom.speed>
<number.mod.bays>
</number.mod.bays>
<bay1.support.set><A/>
</bay1.support.set>
<bay2.support.set><A/>
</bay2.support.set>
<other.mass.storage.set>
<other.mass.storage>
</other.mass.storage>
</other.mass.storage.set>
</cpu>
<audio.video>
 <display.tech><A/>
 </display.tech>
```

```
<d.size.set>
<d.size>13.3
</d.size>
<d.size>14.1
</d.size>
</d.size.set>
<video.bus.type><A/>
</video.bus.type>
<v.memory.installed>
</v.memory.installed>
<v.memory.max>
</v.memory.max>
<v.memory.type><A/>
</v.memory.type>
<d.res><A/>
</d.res>
<color.palette><A/>
</color.palette>
<ex.monitor><A/>
</ex.monitor>
<ex.monitor.max.res><A/>
</ex.monitor.max.res>
<ex.color.palette><A/>
</ex.color.palette>
<sound.support.compat><A/>
</sound.support.compat>
<midi><A/>
</midi>
<full.duplex yesorno="yes"/>
<audio.ports><A/>
</audio.ports>
<internal.speakers><A/>
</internal.speakers>
<builtin.mphone yesorno="yes"/>
<builtin.modem yesorno="yes"/>
<speakerphone yesorno="yes"/>
<voicemail yesorno="yes"/>
<flash.upgrade.modem yesorno="yes"/>
</audio.video>
<portex>
<system.bus.type><A/>
</system.bus.type>
<pcmcia.slots><A/>
</pcmcia.slots>
<cardbus.pccard yesorno="yes"/>
<zv.pccard yesorno="yes"/>
<pds.slots>
</pds.slots>
<ex.scsi.connector><A/>
</ex.scsi.connector>
<serial.ports>
</serial.ports>
<inf.serial.ports>
</inf.serial.ports>
<parallel.port><A/>
</parallel.port>
<extended.parallel.protocols><A/>
```

```
    </extended.parallel.protocols>
    <keyboard.mouse.port.type><A/>
    </keyboard.mouse.port.type>
    <mouse.port.type><A/>
    </mouse.port.type>
    <other.port.set>
    <other.port>
    </other.port>
    </other.port.set>
   </portex>
   <accessory>
    <accessory.purpose>
    </accessory.purpose>
    <accessory.replaces>
    </accessory.replaces>
    <accessory.works.with>
    </accessory.works.with>
   </accessory>
  </product.specifications>

</product.description>
```

```
<!-- thinkcat.xml Version:  0.1 -->
<!-- Purpose:  catalog entry for Ingram Micro demo -->
<!-- Terry Allen  5 Jan 1998 -->
<!-- Copyright 1998 CNgroup, Inc. -->

<?xml version="1.0"?>
<!DOCTYPE catalog.entry SYSTEM "icat.dtd">

<catalog.entry>
<market.participant.info.pointer>
<xll.locator urllink="ingram.xml"/>
</market.participant.info.pointer>
<catalog.entry.id>ingthink770
</catalog.entry.id>
<product.description.info.pointer>
<xll.locator urllink="think.xml"/>
</product.description.info.pointer>
<price.info>TBS
</price.info>
</catalog.entry>
```

```
<!-- ttlattri.mod   Version:  0.21 -->
<!-- Purpose:  group time-to-live attributes -->
<!-- Terry Allen  5 Oct 1997 -->
<!-- Copyright 1997 CNgroup, Inc. -->

<!ENTITY % ttl.attrib
   "commences CDATA #IMPLIED
   expires CDATA #IMPLIED"
>
```

Exhibit I. Glushko, Robert J., Implementing Domain-specific Commerce Languages with a Common Business Library, Slides 29-31 (delivered July 25, 1998) accessed at http://groups.haas.berkeley.edu/citm/conferences/cec/Presentations/Session3/ glushko.pdf on October 26, 2006

Implementing Domain-specific Commerce Languages with a Common Business Library

Dr. Robert Glushko -- Director, Information Engineering

4005 Miranda Avenue, Suite 150

Palo Alto, CA 94304

415.858.7711 main

415.858.4925 fax

www.veosystems.com

info@veosystems.com

Outline of the Talk

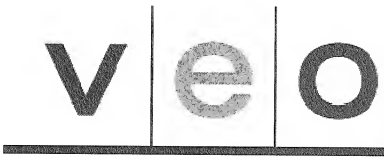
- XML as a technology platform for commerce applications
- Domain-specific commerce languages
- A common business library

veo

About Veo Systems

- **History**
  - 1/97 — for-profit "spin-off" of CommerceNet Consortium called "CNgroup"
  - 9/97 — received multi-million $ award from U.S. Commerce Department ATP to help commercialize "eCo" component-based commerce framework (along with CommerceNet)
  - 4/98 — changed name from CNgroup -Veo
- **Status**
  - Privately held, backed by corporate and VC investors, growing very fast
  - Products later this year

veo



XML as Technology Platform

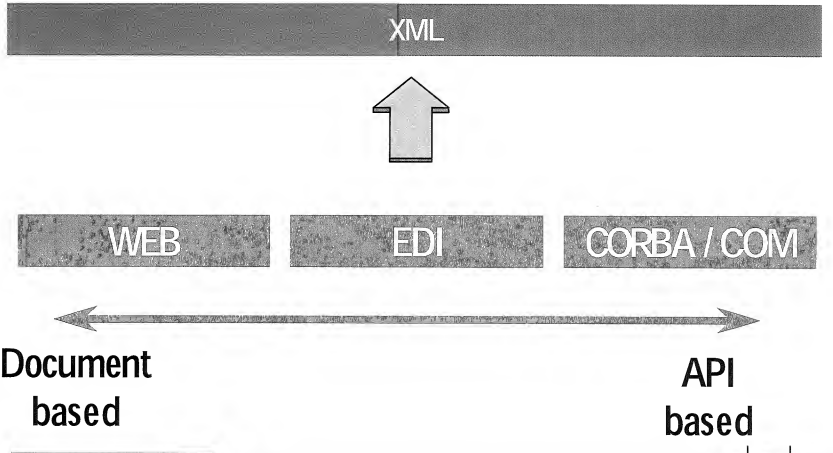
The XML Revolution

- Today's Web sites publish information for people
  - "eyeballs-only" is dominant design perspective
  - hard to search
  - hard to automate processing
    (too much "scraping and hoping")
- Tomorrow's sites will provide information and services for computers (and people)
  - overcomes HTML's inherent limitations
  - enables the new business models of the network economy

XML as Technology Platform

...exchange data in an application and vendor neutral format

...the simplicity of HTML with the precision of APIs



veo

Commerce Networks ⟷ Shared Information Models

- Supply Chains
  - Merchants, distributors, manufacturers, brokers, logistics, shippers
- Real Estate
  - Brokers, banks, escrow, title, inspection, MLS, government agencies, classifieds, loan aggregators
- Securities
  - Brokers, financial advisors, markets, research services, account management
- Travel
  - Hotels, airlines, rental car agencies, travel agents

veo

Laptop Description Seen "By Eye"

Laptop Computer

IBM Thinkpad 560X

233 Mhz

32 Mb

4 Gb

4.1 pounds

$3200

veo

HTML Laptop Description

```
<TITLE>Laptop Computer</TITLE>
<BODY>
<UL>
<LI>IBM Thinkpad 560X
<LI>233 Mhz
 <LI>32 Mb
 <LI>4 Gb
 <LI>4.1 pounds
 <LI>$3200
</UL></BODY>
```

XML Laptop Description

```
<COMPUTER TYPE="LAPTOP">
<MANUFACTURER>IBM</MANUFACTURER>
<LINE>Thinkpad</LINE>
<MODEL>560X</MODEL>
<SPEED UNIT="MHZ">233</SPEED>
 <MEMORY UNIT="MB">32</MEMORY>
 <DISK UNIT="GB">4</DISK>
 <WEIGHT UNIT="POUND">4.1 </WEIGHT>
 <PRICE CURRENCY="USD">3200</PRICE>
</COMPUTER>
```

Smarter Processing Enabled by XML

- Shared schema for laptops, desktops, and towers
- <COMPUTER> provides a logical container for extracted and manipulating product information as a unit
  - Sort by <MANUFACTURER>, <SPEED>, <WEIGHT>, <PRICE>
- Explicit identification of each part enables its automated processing
  - Convert <PRICE> from "USD" to French Francs, Italian Lira, etc.

veo

Airline Schedule Seen "By Eye"

Airline Schedule

Flight Information

United Airlines #200

San Francisco

11:30

Honolulu

2:30

$368.50

HTML Airline Schedule

```
<Title>Airline Schedule</Title>
<Body>
<H2>Flight Information</H2>
<H3>United Airlines #200</H3>
<UL><LI>San Francisco
<LI>11:30
<LI>Honolulu
<LI>2:30
<LI>$368.50
</UL></Body>
```

Airline Schedule in XML

```
<TransportSchedule  Type="Airline">
<Segment Id="United Airlines #200">
 <Origin>San Francisco</Origin>
<DepartTime TZ="PST">11:30 </DepartTime>
 <Destination>Honolulu</Destination>
<ArriveTime TZ="HST"> 2:30 </ArriveTime>
<Price Currency="USD">368.50</Price>
</Segment>
</TransportSchedule>
```


veo

Example: Schema for Transport

Using the same schema for all scheduled transportation services:

<TransportSchedule Type="Airline">
<TransportSchedule Type="Train">
<TransportSchedule Type="Ferry">

An application could create itineraries that involve more than one service by matching on locations and times

Shared Semantics for Time and Location

Shared semantics for location and time in all schemas that need them enables richer "commerce networks" of services:

<TransportSchedule Type="Airline"> ...
<Destination>Honolulu</Destination>

<Accommodation Type="Hotel">...
<Destination>Honolulu</Destination>

<Event Type="Concert">...
<Destination>Honolulu</Destination>

Domain-Specific Commerce Languages

Domain-Specific Languages for Commerce Networks

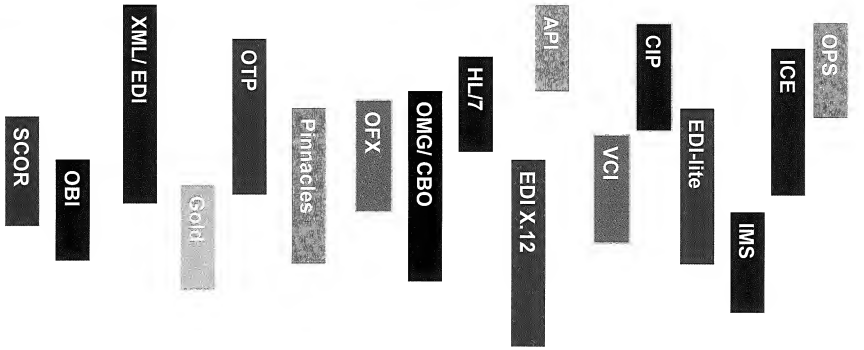
OBI	**Corporate Procurement**	**AMEX, Office Depot, Boise Cascade**
OTP	**Retail Payment**	**Mastercard, Mondex**
OFX / GOLD	**Personal Finance**	**(Intuit, Microsoft), (IBM, 125 Banks)**
ECOM	**Computer Supply Chain**	**Ingram + 24 largest channel players**
ICE	**Content syndication**	**News Corp., Sun, Microsoft, Adobe, Vignette, C/Net**

This list is growing explosively, and all are using XML (or shortly will be)...

XML and Metcalfe's Law

- XML makes it easy to create markup languages
- But the value of a language depends on how many people (or computers) understand it
- How do you encourage and enable others to understand your language?
- The EDI approach:
  - BIG COMPANY: Speak MY language or I won't do business with you!
  - SMALL COMPANY: Yes, master.
- The XML approach:
  - Excuse me, here are the rules of my language if you'd like to speak with me…

Tower of Babel - Stovepipe Protocols



- Delayed time to market
- Redundant development costs
- Limited Interoperability

veo

The Common Business Library

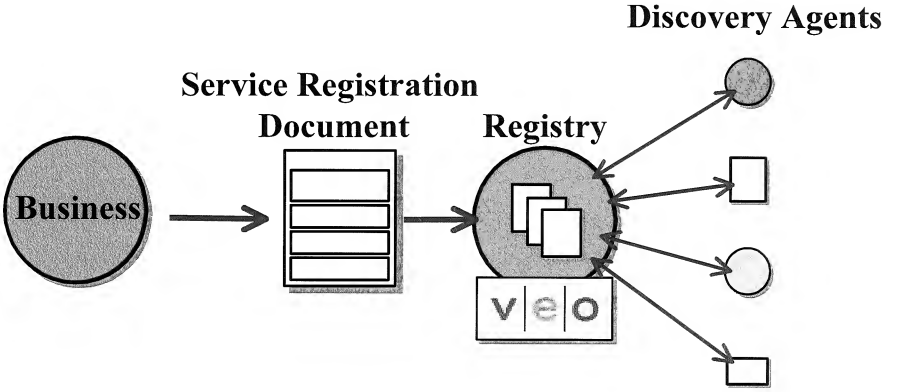
"Loose Coupling" via Shared Document Definitions

- Interconnect business systems and services in terms of the documents they exchange rather than in terms of their application interfaces

- Shared document definitions provide an intuitive framework for specifying the business logic and computations that take place on each end of the exchange.

- Five shared document definitions are implied in these two business rules:
  - if you send me a **request** for a catalog, I will send you a **catalog**
  - if you send me a **purchase order** and I can fulfill it, I will send you a **shipping notice** and an **invoice**

Businesses Publish Services Using XML Documents



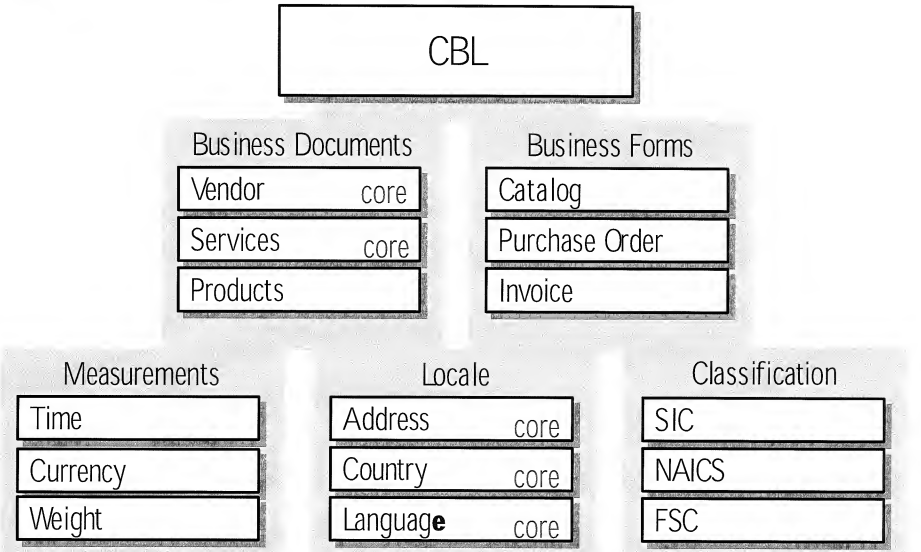
Open Framework For Commerce

veo

The Common Business Library

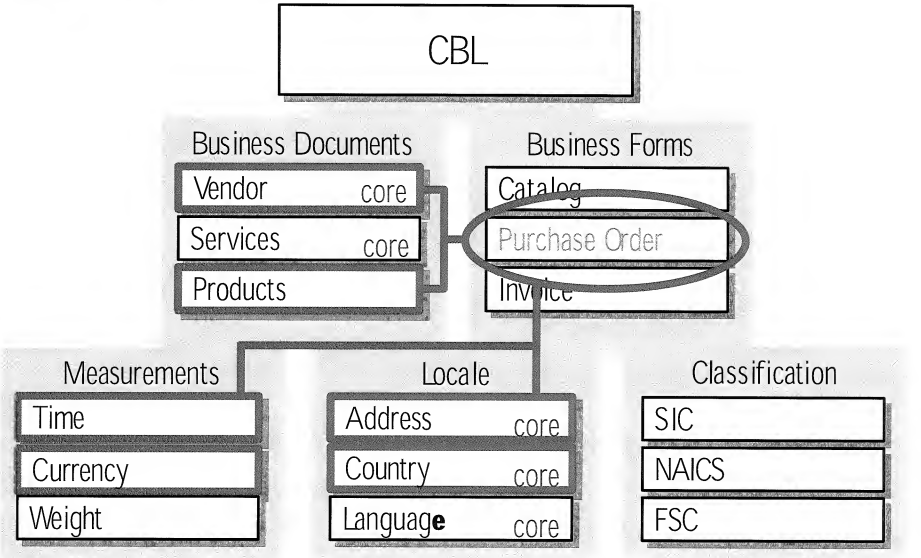
- The functions and information that are common to all business domains, building on existing standards or conventions
- Specifies common semantics, common syntax, and message packaging
- CBL documents are described by XML DTDs to make them "self-descriptive" and validatable
- Complex descriptions and messages can be composed from primitives
- Domain-specific XML applications can be implemented in "native" form or as "hybrids" for maximal interoperability

Building Blocks



veo

Building Blocks

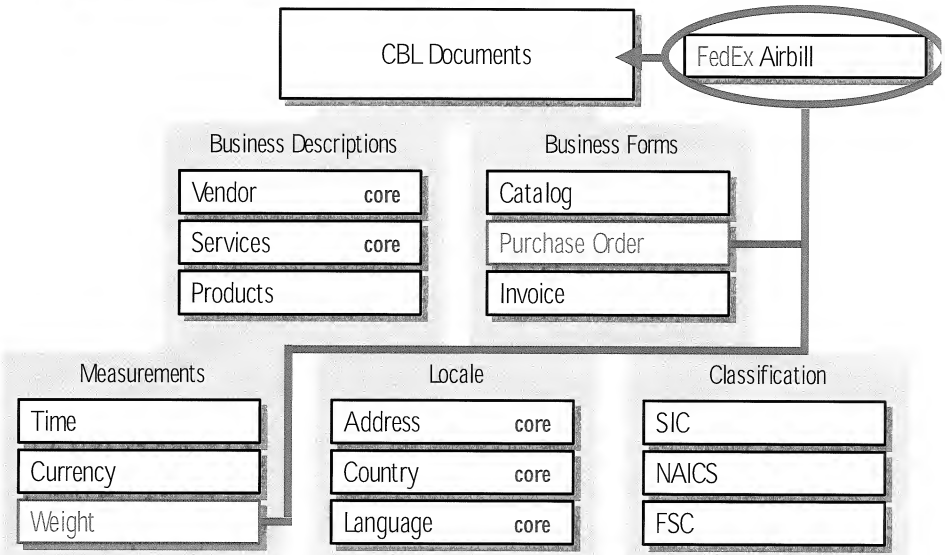


veo

Document Type Definitions and Modules

- addresso.mod, for Geographical Information
- catentry.dtd, for a Simple Catalogue Entry
- contact.mod, for Contact Information
- countrys.mod, for Country Codes
- datetime.mod, for Date and Time markpart.dtd, for Market Participant Information
- markdesc.dtd, for a Market Description
- pay.mod, for Payment Information
- proddesc.dtd, for a Simple Product Description
- shipment.mod, for Shipping Information
- shopcart.dtd, for a Shopping Cart
- transact.dtd, for Transaction Documents

CBL Building Blocks



veo

Business Services Described Using CBL

```
<service>
<service.name>Order Service</service.name>
<service.location>www.veosystems.com/order</service.location>
<service.op>
        <service.op.name>Submit Order</service.op.name>
        <service.op.inputdoc>po.dtd</service.op.inputdoc>
        <service.op.outputdoc>poack.dtd</service.op.outputdoc>
</service.op>
< service.op>
        < service.op.name>Track Order</service.op.name>
        <service.op.inputdoc>request.track.dtd<service.op.inputdoc>
        <service.op.outputdoc>response.track.dtd<service.op.outputdoc>
</service.op>
</service>
```

CBL Status

- CBL v1.0 contains a few dozen DTDs and modules developed from analysis of ISO, ANSI X.12, other standards
- CBL currently being used by Veo Systems in demonstration applications (Project Seitai, GSA catalog interoperability)
- CBL to be starting "fodder" for CommerceNet-sponsored WG to develop open framework for interoperability of domain-specific commerce languages (just getting under way)

The Economist

"Untangling the Web"

25 April 1998

.."But the biggest role that XML is expected to play is in integrating the way that existing paper documents -- invoices, loan applications, contracts, insurance claims, you name it are exchanged between organizations around the world. Imagine what the world would be like if one company's computer system could automatically read any other organization's documents - and make complete sense of them? This is the goal that the technique known as EDI has struggled, unsuccessfully, to achieve for years. Though efforts have barely begun, there is a chance that XML could actually make that happen. If it did, business on the Web could run riot."

Exhibit J. Excerpts from W3C "note" WSDL version 1.1 (March 15, 2001) accessed at http://www.w3.org/TR/wsdl

W3C

Web Services Description Language (WSDL) 1.1

W3C Note 15 March 2001

This version:
http://www.w3.org/TR/2001/NOTE-wsdl-20010315
Latest version:
http://www.w3.org/TR/wsdl
Authors (alphabetically):
Erik Christensen, Microsoft
Francisco Curbera, IBM Research
Greg Meredith, Microsoft
Sanjiva Weerawarana, IBM Research

Abstract

WSDL is an XML format for describing network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information. The operations and messages are described abstractly, and then bound to a concrete network protocol and message format to define an endpoint. Related concrete endpoints are combined into abstract endpoints (services). WSDL is extensible to allow description of endpoints and their messages regardless of what message formats or network protocols are used to communicate, however, the only bindings described in this document describe how to use WSDL in conjunction with SOAP 1.1, HTTP GET/POST, and MIME.

Status

This document is a submission to the World Wide Web Consortium (see Submission Request, W3C Staff Comment) as a suggestion for describing services for the W3C XML Activity on XML Protocols. For a full list of all acknowledged Submissions, please see Acknowledged Submissions to W3C.

This draft represents the current thinking with regard to descriptions of services within Ariba, IBM and Microsoft. It consolidates concepts found in NASSL, SCL, and SDL (earlier proposals in this space).

This document is a NOTE made available by the W3C for discussion only. Publication of this Note by W3C indicates no endorsement by W3C or the W3C Team, or any W3C Members. W3C has had no editorial control over the preparation of this Note. This document is a work in progress and may be updated, replaced, or rendered obsolete by other documents at any time.

A list of current W3C technical documents can be found at the Technical Reports page.

Table of Contents

1. Introduction

As communications protocols and message formats are standardized in the web community, it becomes increasingly possible and important to be able to describe the communications in some structured way. WSDL addresses this need by defining an XML grammar for describing network services as collections of communication endpoints capable of exchanging messages. WSDL service definitions provide documentation for distributed systems and serve as a recipe for automating the details involved in applications communication.

A WSDL document defines **services** as collections of network endpoints, or **ports**. In WSDL, the abstract definition of endpoints and messages is separated from their concrete network deployment or data format bindings. This allows the reuse of abstract definitions: **messages**, which are abstract descriptions of the data being exchanged, and **port types** which are abstract collections of **operations**. The concrete protocol and data format specifications for a particular port type constitutes a reusable **binding**. A port is defined by associating a network address with a reusable binding, and a collection of ports define a service. Hence, a WSDL document uses the following elements in the definition of network services:

- **Types**– a container for data type definitions using some type system (such as XSD).
- **Message**– an abstract, typed definition of the data being communicated.
- **Operation**– an abstract description of an action supported by the service.
- **Port Type**–an abstract set of operations supported by one or more endpoints.
- **Binding**– a concrete protocol and data format specification for a particular port type.
- **Port**– a single endpoint defined as a combination of a binding and a network address.
- **Service**– a collection of related endpoints.

These elements are described in detail in Section 2. It is important to observe that WSDL does not introduce a new type definition language. WSDL recognizes the need for rich type systems for describing message formats, and supports the XML Schemas specification (XSD) [11] as its canonical type system. However, since it is unreasonable to expect a single type system grammar to be used to describe all message formats present and future, WSDL allows using other type definition languages via extensibility.

In addition, WSDL defines a common **binding** mechanism. This is used to attach a specific protocol or data format or structure to an abstract message, operation, or endpoint. It allows the reuse of abstract definitions.

In addition to the core service definition framework, this specification introduces specific **binding extensions** for the following protocols and message formats:

- SOAP 1.1 (see Section 3)
- HTTP GET / POST (see Section 4)
- MIME (see Section 5)

Although defined within this document, the above language extensions are layered on top of the core service definition framework. Nothing precludes the use of other binding extensions with WSDL.

1.2 WSDL Document Example

The following example shows the WSDL definition of a simple service providing stock quotes. The service supports a single operation called GetLastTradePrice, which is deployed using the SOAP 1.1 protocol over HTTP. The request takes a ticker symbol of type string, and returns the price as a float. A detailed description of the elements used in this definition can be found in Section 2 (core language) and Section 3 (SOAP binding).

This example uses a fixed XML format instead of the SOAP encoding (for an example using the SOAP encoding, see Example 4).

Example 1 SOAP 1.1 Request/Response via HTTP

```
<?xml version="1.0"?>
<definitions name="StockQuote"

targetNamespace="http://example.com/stockquote.wsdl"
          xmlns:tns="http://example.com/stockquote.wsdl"
          xmlns:xsd1="http://example.com/stockquote.xsd"
          xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
          xmlns="http://schemas.xmlsoap.org/wsdl/">

    <types>
        <schema targetNamespace="http://example.com/stockquote.xsd"
                xmlns="http://www.w3.org/2000/10/XMLSchema">
            <element name="TradePriceRequest">
                <complexType>
                    <all>
                        <element name="tickerSymbol" type="string"/>
                    </all>
                </complexType>
            </element>
            <element name="TradePrice">
                <complexType>
                    <all>
                        <element name="price" type="float"/>
                    </all>
                </complexType>
            </element>
        </schema>
    </types>

    <message name="GetLastTradePriceInput">
        <part name="body" element="xsd1:TradePriceRequest"/>
    </message>

    <message name="GetLastTradePriceOutput">
        <part name="body" element="xsd1:TradePrice"/>
    </message>

    <portType name="StockQuotePortType">
        <operation name="GetLastTradePrice">
            <input message="tns:GetLastTradePriceInput"/>
            <output message="tns:GetLastTradePriceOutput"/>
        </operation>
    </portType>
```

```
    <binding name="StockQuoteSoapBinding" type="tns:StockQuotePortType">
        <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
        <operation name="GetLastTradePrice">
            <soap:operation soapAction="http://example.com/GetLastTradePrice"/>
            <input>
                <soap:body use="literal"/>
            </input>
            <output>
                <soap:body use="literal"/>
            </output>
        </operation>
    </binding>

    <service name="StockQuoteService">
        <documentation>My first service</documentation>
        <port name="StockQuotePort" binding="tns:StockQuoteBinding">
            <soap:address location="http://example.com/stockquote"/>
        </port>
    </service>

</definitions>
```

1.2 Notational Conventions

1. The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119 [2].

2. The following namespace prefixes are used throughout this document:

prefix	namespace URI	definition
wsdl	http://schemas.xmlsoap.org/wsdl/	WSDL namespace for WSDL framework.
soap	http://schemas.xmlsoap.org/wsdl/soap/	WSDL namespace for WSDL SOAP binding.
http	http://schemas.xmlsoap.org/wsdl/http/	WSDL namespace for WSDL HTTP GET & POST binding.
mime	http://schemas.xmlsoap.org/wsdl/mime/	WSDL namespace for WSDL MIME binding.
soapenc	http://schemas.xmlsoap.org/soap/encoding/	Encoding namespace as defined by SOAP 1.1 [8].
soapenv	http://schemas.xmlsoap.org/soap/envelope/	Envelope namespace as defined by SOAP 1.1 [8].
xsi	http://www.w3.org/2000/10/XMLSchema-instance	Instance namespace as defined by XSD [10].
xsd	http://www.w3.org/2000/10/XMLSchema	Schema namespace as defined by XSD [10].
tns	(various)	The "this namespace" (tns) prefix is used as a convention to refer to the current document.
(other)	(various)	All other namespace prefixes are samples only. In particular, URIs starting with "http://example.com" represent some application-dependent or context-dependent URI [4].

3. This specification uses an **informal syntax** to describe the XML grammar of a WSDL document:

- The syntax appears as an XML instance, but the values indicate the data types instead of values.
- Characters are appended to elements and attributes as follows: "?" (0 or 1), "*" (0 or more), "+" (1 or more).
- Elements names ending in "..." (such as <element.../> or <element...>) indicate that elements/attributes irrelevant to the context are being omitted.
- Grammar in bold has not been introduced earlier in the document, or is of particular interest in an example.
- <-- extensibility element --> is a placeholder for elements from some "other" namespace (like ##other in XSD).